In The Name Of Allah

Book:

Lets Learn JaVa

By:

Ahmed Salah
Faculty Of Engineering
Electrical Eng. Department
Power & Machines Section

Email:

d1e.h4rd@yahoo.com

المحتويات:

التعريف بلغه الجافا:

هي لغة حديثة الى حد ما حيث تم انتاجها عام ١٩٩٥م لتناسب التطبيقات الحديثة،وهي تناسب تطبيقات الإنترنت حيث أصبحت قلب برمجة الإنترنت بما توفره من إمكانيات .

==========

مواصفات لغه الجافا:

١- لغة برمجية تدعم مبادئ الكائنية .
٢- لها بيئة تشغيل خاصة بها.
٣- تقوم على لغة C و C++.
٤- تعمل على معظم نظم التشغيل.

==========

اهم مميزات لغه الجافا:

١- إضافة الحركة والصوت إلى صفحات الويب.
٢- كتابة الألعاب والبرامج المساعدة.
٣- إنشاء برامج ذات واجهة مستخدم رسومية.

٤- تصميم برمجيات تستفيد من كل مميزات الانترنت.
٥- كتابة برامج تعليمية للإنترنت عبر برمجيات المحاكاة الحاسوبية للتجارب العلمية وبرمجيات الفصول الافتراضية للتعليم الإلكتروني والتعليم عن بعد.

===========

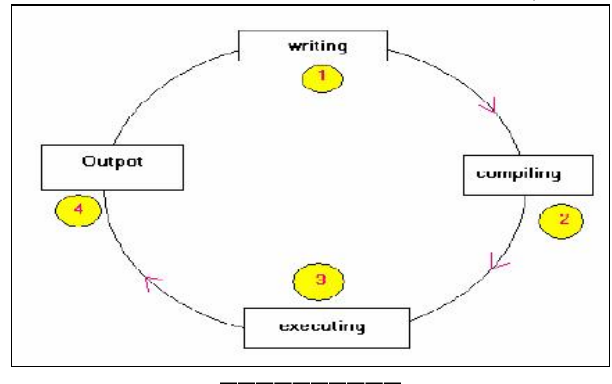
ملاحظات خاصه بهذه اللغه:

١- اللغه حساسه للاحرف الصغيره والكبيره.
٢- يمكن استخدام اسطر فارغه حيث انها لا تؤثر فى البرمجه.
٣- يجب حفظ الملف الخاص بالبرنامج بامتداد java .

===========

المراحل التى يمر بها بناء اى برنامج بالجافا:

١- Writing يقصد كتابه البرنامج.
٢- Compiling يقصد بها معاينه البرنامج وتصحيح ما به من اخطاء.
٣- Executing تنفيذ البرنامج.
٤- Output اظهار المخرجات.



===========

البرنامج المستخدم وملحقاته:

لابد من تنصيب ما يلى من برامج،وهما كالتالى:

اولا:

NetBeans IDE 6.9.1

ثانيا:

JDK

===========

مكونات لغه الجافا:

١- الرموز الخاصه.
٢- الاختصارات المسانده.
٣- الكلمات المحجوزه.
٤- الحروف الابجديه.
٥- الارقام العدديه.

اولا: الحروف الابجديه:

وهى الحروف الكبيره من A حتى Z ،وايضا الحروف الصغيره من a حتى z .

ثانيا: الارقام العدديه:

وهى الارقام من 0 حتى 9 .

ثالثا: الرموز الخاصه:

وهى العلامات او الرموز التى ليست بارقام او حروف كعلامه الجمع +،وعلامه الطرح –،وعلامه الضرب * ،وعلامه القسمه /،وعلامه المساواه =.....الخ.

رابعا: الاختصارات المسانده:

ونذكر منها على سبيل المثال لا الحصر:

\n	سطر جديد.
\t	التقدم سبع مسافات.
\b	العوده مسافه للخلف.

خامسا: الكلمات المحجوزه:

ويشترط ان لا نستخدمها الا لوظيفتها المخصصه،ولا يجوز ان تكون اسماء لمتغيرات، ونذكر منها على سبيل المثال لا الحصر:

Return ، void ، for ، short ، if ....الخ

===========

انواع المتغيرات:

اولا: المتغير الصحيح integer :

متغيرات عدديه صحيحه وقد تكون موجبه او سالبه ولا يحتوى على نقطه عشريه او اسيه ومثال له 4 , 70 , -6 ,.....الخ.

ثانيا: المتغير العشرى float :

متغيرات عدديه عشرات ذات الفاصله العائمه ومثال له 5.0 , 4.0 , -0.1 , -0.8 ,......الخ.

ثالثا: المتغير الحرفى Char :

متغيرات عباره عن حرف واحد فقط وبين اشارتى اقتباس مفرده ' ' ومثال له a , b , c .....الخ.

رابعا: متغير السلسله string :

يحتوى هذا النوع من البيانات على اى عدد من الرموز او الفراغات وتكون محصوره بين علامتى تنصيص " ".

خامسا: متغير الصواب والخطأ Boolean :

ويأخذ قيمتين اما الصواب true او الخطأ false .

سادسا: المتغير المزدوج Double :

هي نفس المتغيرات العددية العشرية ولكن يمكن تمثيلها إلى خمسة عشر خانه،اى ان قيم هذا النوع تكون اكبر بكثير من قيم النوع float مثلا 13.40 , -77.987 , 13.15 , 55.98 ......الخ

===========

الصيغة العامه لتعريف متغير ما :-

Datatype variable_name = value

حيث:-

Datatype نوع المتغير سواء كان int او float او char او double ..

Variable_name اسم المتغير وذلك يخضع لنوع المتغير السابق ادخاله ..

Value القيمه المراد تخزينها بالمتغير..

امثله:-

```
Bollean x = true
long z = 123432l
float y = 1.23f
char z = 'A'
```

لاحظ عند تعريف متغير من النوع long قمنا بكتابه الحرف l بعد قيمته،وبالمثل عند تعريف متغير من النوع float قمنا بكتابه الحرف f بعد قيمته.

===========

التعليقات:

اذا اردت كتابه تعليق على اى سطر من سطر البرنامج هناك طريقتين وهما كالتالى:

الاولى :

```
// Your Comment
```

والثانيه:

```
/*
Your comment
*/
```

ولن يلتفت اليه البرنامج او يضعه فى حساباته.

============

الحلقات التكراريه والجمل الشرطيه:

الصيغه الاولى:

```
(1)

IF ( الشرط )
{
    ═══ ; code

}
```

الصيغه الثانيه:

```
(2)

IF ( الشرط )
{
    ═══ ; code

}
Else
{
    ═══ ; code

}
```

الصيغه الثالثه:

```
(3)

IF ( الشرط )
{
    ═══ ; code

}
Else IF ( الشرط )
{
    ═══ ; code

}
Else
{
    ═══ ; code

}
```

الصيغه الرابعه:

```
switch ( Test )
{
case value1 :
    ═══ ; code
    break ;

case value2 :
    ═══ ; code
    break ;
      :
      :
default :
    ═══ ; code
    break ;
}
```

الصيغه الخامسه:

```
for ( نقطة البداية ; الشرط ; مقدار الزيادة )
{
    ; code
}
```

الصيغه السادسه:

```
while ( الشرط )
{
    ;   code
        +
        index
}
```

الصيغه السابعه:

```
do
{
    ;   code
        +
        index
} while ( الشرط ) ;
```

============

اوامر قد تفيدك:-

system.out.println( )

يستخدم هذا الامر لطباعه شئ ما ،فيطبعه ثم ينزل سطر.

system.out.print( )

يستخدم هذا الامر لطباعه شئ ما ،فيطبعه بدون ان ينزل سطر.

system.exit(0)

يستخدم هذا الامر للخروج من التطبيق.

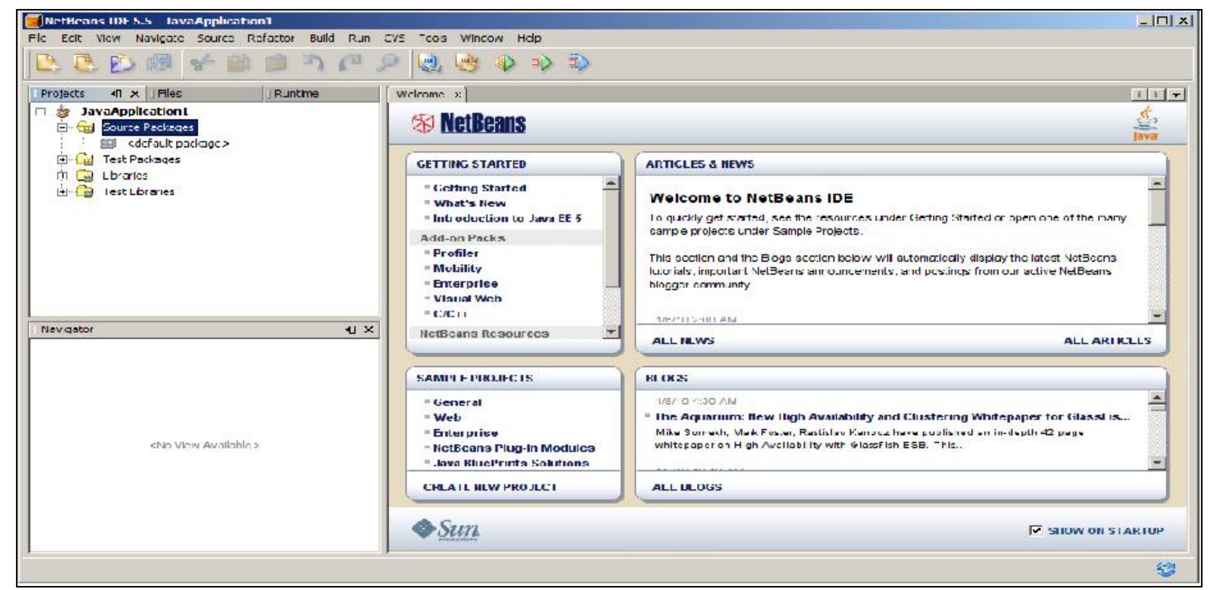
============

طريقه فتح مشروع جديد:

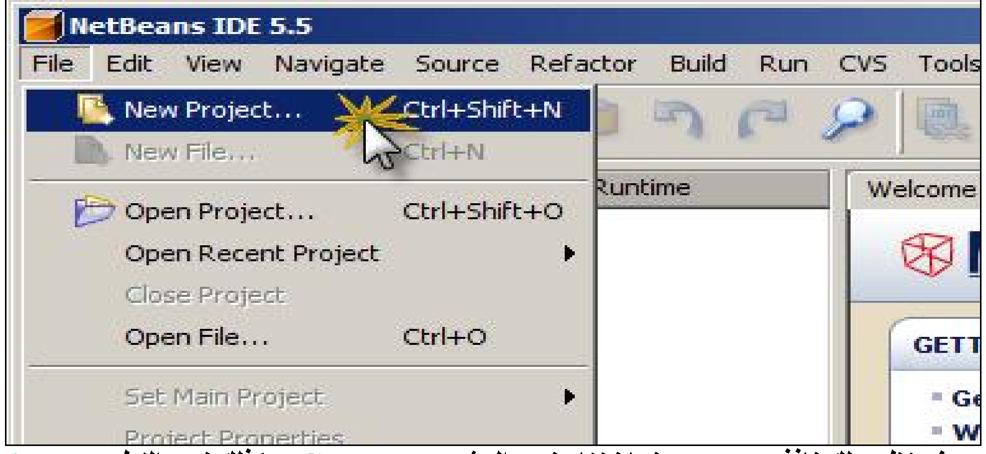
اولا: من على سطح المكتب نقوم بالضغط مرتين على ايقونه محرر لغه الجافا كالتالى:

NetBeans 5.5

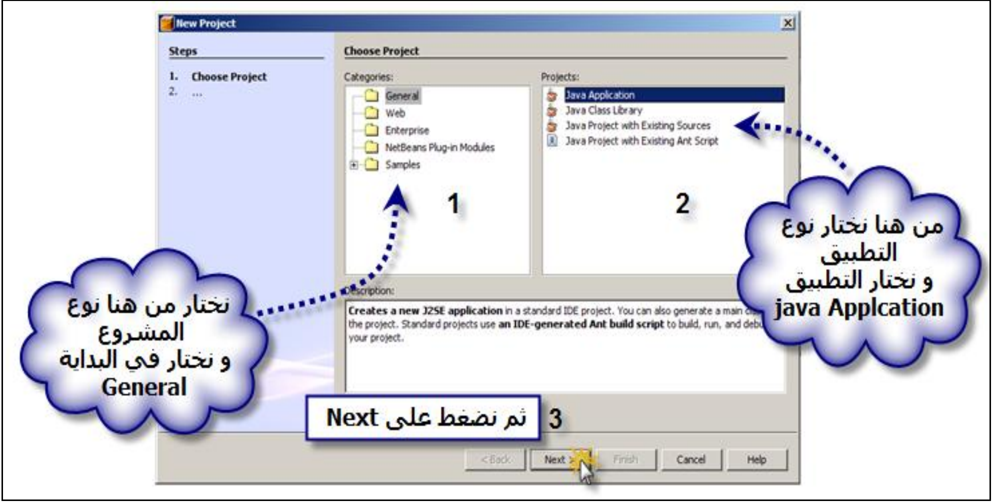
ثانيا: سوف تظهر لك النافذه التاليه للمحرر بالشكل التالى:



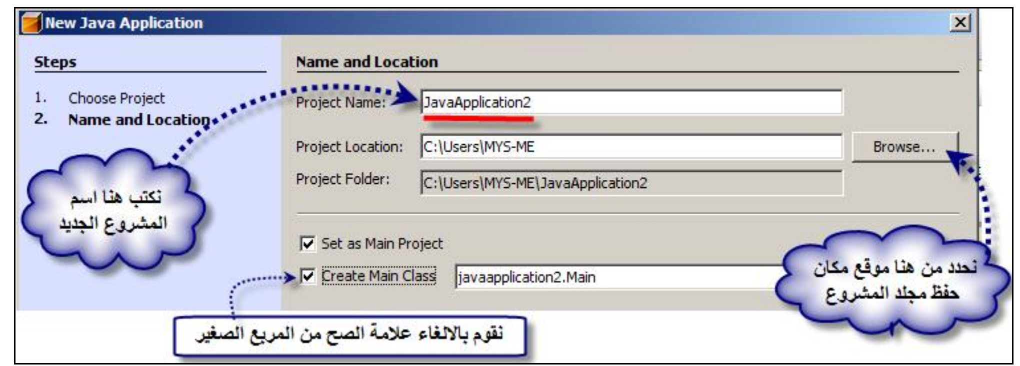
ثالثا: من قائمه **file** سوف نختار **new project** كالتالى:



رابعا: سوف تظهر لك نافذه جديده ،ومنها نختار نوع المشروع **General** ،وكذلك نوع التطبيق **java application**،كالتالى:



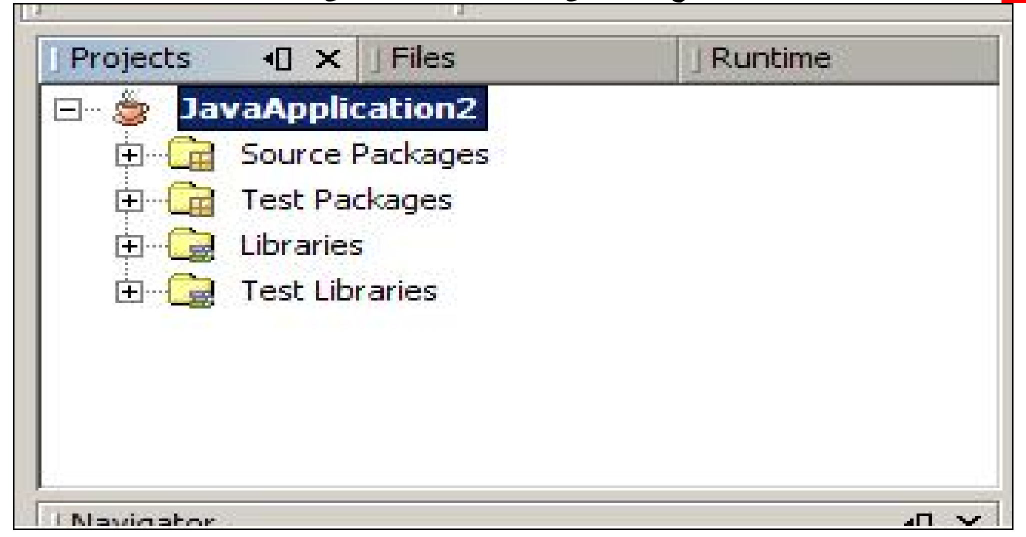
خامسا: سوف تظهر لك نافذه اخرى جديده،ومنها نختار اسم المشروع،وموقعه،كالتالى:



سادسا: ثم الضغط على **finish** .

Finish

سابعا: سوف تظهر لك بيانات المشروع الجديد فى النافذه الرئيسيه كالتالى:



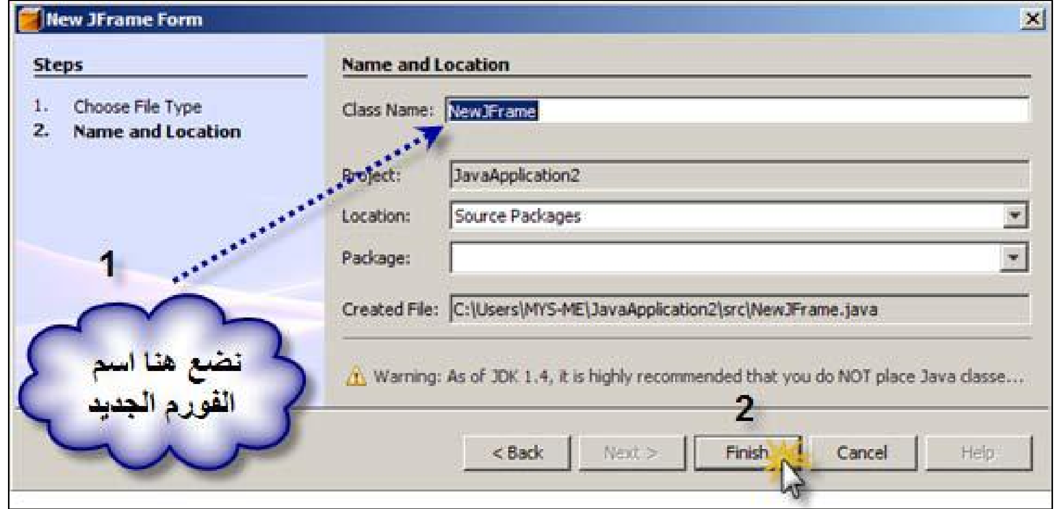
============

ادراج form جديد لبدأ عمليه تصميم المشروع:

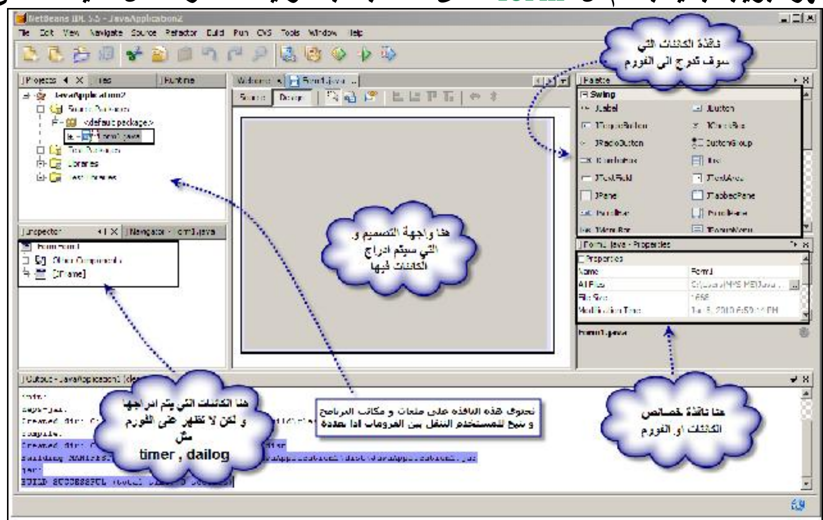
اولا:



ثانيا:



ثالثا: سوف يظهر تبويب جديد باسم ال form الذى قمت بكتابته وايضا اطار العمل عليه،كالتالى:



============

تنفيذ البرنامج:

من اجل تنفيذ البرنامج الذى قمنا بتصميمه،فاننا سوف نتبع احدى الطرق التاليه:

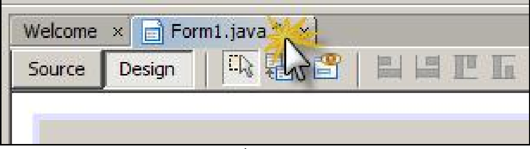
الاولى: نضغط على الزر f5 من لوحه المفاتيح.

الثانيه: نضغط على زر run الموجود اعلى البرنامج.

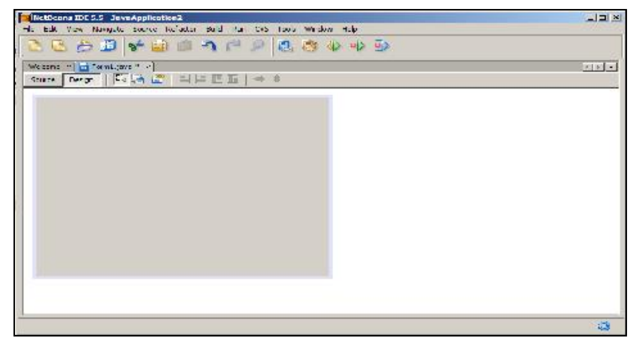
============

تكبير وتصغير نافذه ال form :

عند النقر المزدوج على تبويب نافذه ال form كالتالى:



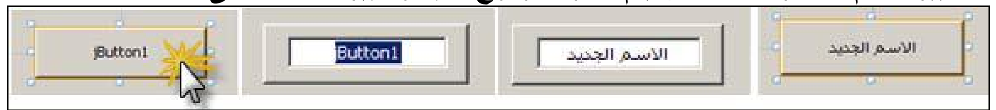
فانه سوف يتم تكبير نافذه ال form واظهارها بالكامل وذلك من اجل تحكم اكبر بشكل ال form ،وايضا من اجل الرجوع الى الوضعيه الاصغر يتم النقر المزدوج على التبويب نفسه كالتالى:



============

تغيير الاسم الظاهرى لاى كائن:

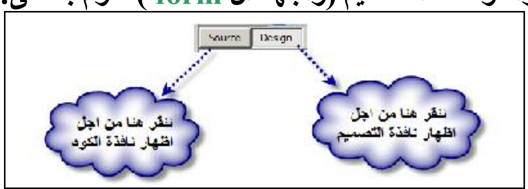
من اجل تغيير الاسم الظاهرى لاى كائن يتم النقر المزدوج عليه وتغيير اسمه كالتالى:



============

التنقل بين النوافذ:

من اجل التنقل بين نافذه الاكواد ونافذه التصميم (واجهه ال **form**) نقوم بالتالى:



============

اختيار حدث لاى كائن وكتابه الكود بداخله:

من اجل اختيار حدث لكائن ما وكتابه الكود بداخله نتبع ما يلى:

١- نحدد الكائن المراد التعديل او العمل عليه.
٢- ننقر بزر الماوس الايمن عليه.
٣- سوف تظهر لنا قائمه كبيره نختار كلمه **events** .
٤- سوف تظهر لنا قائمه اخرى كبيره نختار منها تصنيف الحدث وليكن **mouse** .
٥- سوف تظهر لنا قائمه اخرى كبيره نختار منها الحدث المطلوب وليكن **mouse clicked** .

وكل هذا موضح بالشكل التالى:



============

الشفرات اللازمه لبناء اى form :

عند فتح نافذه الكود فاننا نلاحظ بدايه الكود بهذه السطور:

```
public class Form1 extends javax.swing.JFrame {

    /** Creates new form Form1 */
    public Form1() {
        initComponents();
    }
```

============

 تطبيقات بسيطه:

✓ التطبيق الاول: تصميم برنامج يقوم بجمع عددين واظهار النتيجه!!

الكود:

```
double No1 ;
double No2 ;
double Res ;
No1 = Double.parseDouble(jTextField1.getText( ) ) ;
No2 = Double.parseDouble(jTextField2.getText( ) ) ;
Res = No1 + No2 ;
jLabel1.setText("" + Res) ;
```

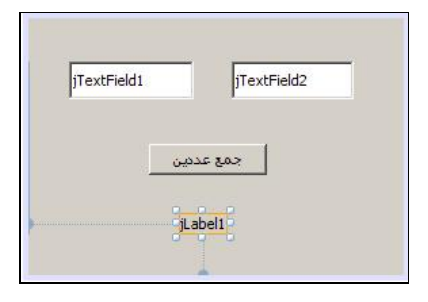
موقع الكود:

نكتب الكود السابق فى الكائن(جمع عددين).

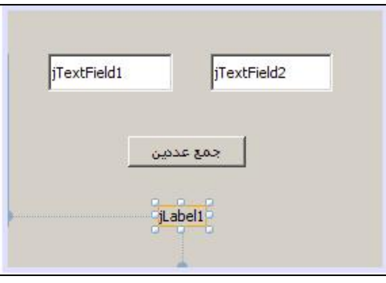
الشرح:

السطر الاول:قمنا بتعريف متغير جديد من النوع double يدعى No1 وهو يمثل الرقم الاول.
السطر الثانى:قمنا بتعريف متغير جديد من النوع double يدعى No2 وهو يمثل الرقم الثانى.
السطر الثالث:قمنا بتعريف متغير جديد من النوع double يدعى Res وهو يمثل الناتج.
السطر الرابع:تم اخذ النص الذى سيدخله المستخدم ل jTextField1 وذلك باستخدام الدالة getText( )،ومن ثم تم تحويل البيانات او الارقام المدخله من نصيه الى رقميه وذلك باستخدام الدالة parseDouble ،ومن ثم ادراج الرقم المدخل داخل المتغير No1 .
السطر الخامس:بالمثل كما سبق ولكن فى النهايه سيتم ادراج الرقم المدخل المتغير No2 .
السطر السادس:عمليه جمع المتغييرين وادراج القيمه الجديده فى المتغير Res .
السطر السابع:يقوم بادراج القيمة الموجوده في المتغير Res الى الكائن jLabel1 ،وذلك باستخدام الدالة setText .

التصميم:



البرنامج:



===========

✓ التطبيق الثانى: تصميم برنامج لاظهار نتيجه الطالب بحيث عندما يدخل المستخدم درجه الطالب يكون الخرج اما ناجح او راسب!!

الكود:

```
float m;
m = Float.parseFloat(jTextField1.getText( ) );
if (m>=0 & m<50)
{jLabel1.setText("failed";}
else if (m>50 & m<=100)
{jLabel1.setText("passed";}
else if (m==50)
{jLabel1.setText("passed";}
else
{jLabel1.setText("Error!!";}
```

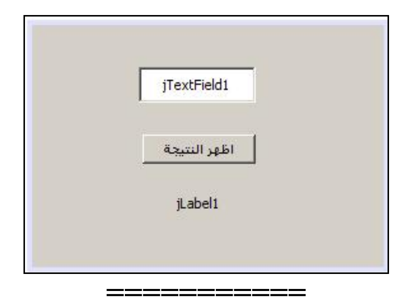
موقع الكود:

نكتب الكود السابق فى الكائن(اظهر النتيجه).

الشرح:

فرضنا ان الدرجه النهائيه للاختبار هى ١٠٠ درجه.
وعليه فاذا حصل الطالب على درجه من ٥٠ الى ١٠٠ يعتبر ناجح passed!!،واذا حصل الطالب على درجه من ١ الى ٤٩ يعتبر راسب failed!!،واذا ادخل المستخدم قيم ما بخلاف ما سبق يطبع البرنامج كلمه خطأ Error!! .
وقد استخدمنا لتحقيق ما سبق من شروط جمله if الشرطيه.
السطر الاول: قمنا بتعريف متغير جديد من النوع float يدعى m وهو يمثل الدرجه التى سيدخلها المستخدم.
السطر الثانى: تم اخذ النص الذى سيدخله المستخدم ل jTextField1 وذلك باستخدام الداله getText( )،ومن ثم تم تحويل البيانات او الارقام المدخله من نصيه الى رقميه وذلك باستخدام الداله parseFloat ،ومن ثم ادراج الرقم المدخل داخل المتغير m .
من السطر الثالث حتى الاخير:جمله if الشرطيه.

التصميم:



============

✓ التطبيق الثالث: تصميم برنامج يقوم بتحديد تقدير الطالب من (ممتاز) حتى (ضعيف) حسب الدرجه المدخله اليه من قبل المستخدم!!

الكود:

```
float m;
m = Float.parseFloat(jTextField1.getText( ) );
if (m>=0 & m<50)
{jLabel1.setText("";}
else if (m>=50 & m<65)
{jLabel1.setText("";}
else if (m>=65 & m<80)
{jLabel1.setText("";}
else if (m>=80 & m<90)
{jLabel1.setText("";}
else if (m>=90 & m<=100)
{jLabel1.setText("";}
else
```

```
{jLabel1.setText("";}
```

موقع الكود:

نكتب الكود السابق فى الكائن(اظهر النتيجه).

الشرح:

فرضنا ان الدرجه النهائيه للاختبار هى ١٠٠ درجه.
وعليه:
فاذا حصل الطالب على درجه من ٠ الى ٥٠ يكون تقديره ضعيف .
واذا حصل الطالب على درجه من ٥٠ الى ٦٥ يكون تقديره مقبول .
واذا حصل الطالب على درجه من ٦٥ الى ٨٠ يكون تقديره جيد .
واذا حصل الطالب على درجه من ٨٠ الى ٩٠ يكون تقديره جيد جدا .
واذا حصل الطالب على درجه من ٩٠ الى ١٠٠ يكون تقديره ممتاز .
اذا ادخل المستخدم قيم ما بخلاف ما سبق يطبع البرنامج كلمه خطأ .
وقد استخدما لتحقيق ما سبق من شروط جمله if الشرطيه.
السطر الاول: قمنا بتعريف متغير جديد من النوع float يدعى m وهو يمثل الدرجه التى سيدخلها المستخدم.
السطر الثانى: تم اخذ النص الذى سيدخله المستخدم ل jTextField1 وذلك باستخدام الداله getText( )،ومن ثم تم تحويل البيانات او الارقام المدخله من نصيه الى رقميه وذلك باستخدام الداله parseFloat ،ومن ثم ادراج الرقم المدخل داخل المتغير m .
من السطر الثالث حتى الاخير:جمله if الشرطيه.

التصميم:

============

✓ التطبيق الرابع: تصميم برنامج لاظهار الارقام بالحروف عندما يقوم المستخدم بادخال رقم ما
مثلا:
١ـ اذا ادخل المستخدم رقم ١ يظهر لنا كلمه one .
٢ـ اذا ادخل المستخدم رقم ٢ يظهر لنا كلمه two .
وهكذا!!وليكن حتى رقم ٥ فقط!!

الكود:

```
int n ;
n = Integer.parseInt(jTextField1.getText( )) ;
switch (n)
{
case 1 :
jLabel1.setText("one") ;
break ;
case 2 :
jLabel1.setText("two") ;
break ;
case 3 :
jLabel1.setText("three") ;
break ;
case 4 :
jLabel1.setText("four") ;
break ;
```

```
case 5 :
jLabel1.setText("five") ;
break ;
default:
jLabel1.setText("enter the correct number!!") ;
break ;
}
```

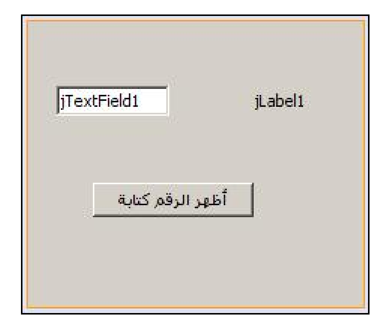
موقع الكود:

نكتب الكود السابق فى الكائن(اظهر الرقم كتابه).

الشرح:

السطر الاول: قمنا بتعريف متغير جديد من النوع int يدعى n وهو يمثل الدرجه التى سيدخلها المستخدم.
السطر الثانى: تم اخذ النص الذى سيدخله المستخدم ل jTextField1 وذلك باستخدام الداله getText() ومن ثم تم تحويل البيانات او الارقام المدخله من نصيه الى رقميه وذلك باستخدام الداله parseInt ، ومن ثم ادراج الرقم المدخل داخل المتغير n .
من السطر الثالث حتى الاخير:جمله switch الشرطيه.

التصميم:



البرنامج:



=============

✓ التطبيق الخامس: تصميم برنامج لاضافه الاعداد من واحد الى عشره الى كائن من نوع jComboBox !!

الكود:

```
jComboBox1.removeAllItems( ) ;
for (int i=1 ; i<=10 ; i++)
{
jComboBox1.addItem(i) ;
}
```

موقع الكود:

نكتب الكود السابق فى الكائن(ادرج الارقام).

الشرح:

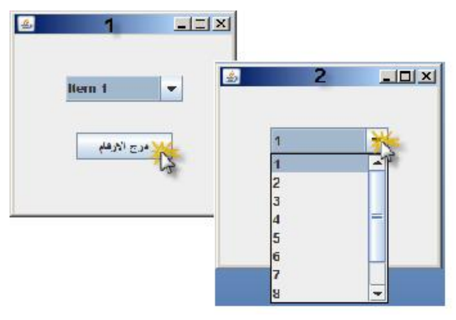
السطر الاول:يقوم بتفريغ الكائن jComboBox1 من جميع العناصر وقد تم ذلك باستخدام الداله removeAllItems( ).
السطر الثانى:الحلقه التكراريه for والتى تحتوى على ثلاث اجزاء:
الجزء الاول: ينفذ قبل اول لفه فقط وهو int i=1 .
الجزء الثانى: ينفذ قبل كل لفه ويسمى الشرط وهو i<=10 .

الجزء الثالث: ينفذ بعد كل لفه ويسمى الخطوه وهو ++i .
السطر الرابع:يقوم باضافه العناصر الى الكائن jComboBox1 وقد تم ذلك باستخدام الداله ( )addItem .

التصميم:



البرنامج:



=============

✓ التطبيق السادس: تصميم برنامج يقوم بضرب عددين واظهار النتيجه!!

الكود:

```
double n1 ;
double n2 ;
double res ;
n1 = Double.parseDouble(jTextField1.getText( ) ) ;
n2 = Double.parseDouble(jTextField2.getText( ) ) ;
res = n1 * n2 ;
jLabel1.setText("" + res) ;
```

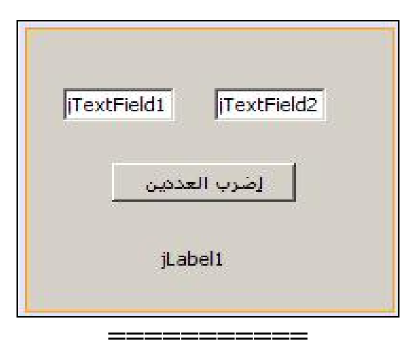
موقع الكود:

نكتب الكود السابق فى الكائن(ضرب عددين).

الشرح:

السطر الاول:قمنا بتعريف متغير جديد من النوع double يدعى n1 و هو يمثل الرقم الاول.
السطر الثانى:قمنا بتعريف متغير جديد من النوع double يدعى n2 و هو يمثل الرقم الثانى.
السطر الثالث:قمنا بتعريف متغير جديد من النوع double يدعى res و هو يمثل الناتج.
السطر الرابع:تم اخذ النص الذى سيدخله المستخدم ل jTextField1 وذلك باستخدام الداله getText( )،ومن ثم تم تحويل البيانات او الارقام المدخله من نصيه الى رقميه وذلك باستخدام الداله parseDouble ،ومن ثم ادراج الرقم المدخل داخل المتغير n1 .
السطر الخامس:بالمثل كما سبق ولكن فى النهايه سيتم ادراج الرقم المدخل المتغير n2 .
السطر السادس:عمليه ضرب المتغيرين وادراج القيمه الجديده فى المتغير res .
السطر السابع:يقوم بادراج القيمة الموجوده في المتغير res الى الكائن jLabel1 ،وذلك باستخدام الداله setText .

التصميم:



============

✓ التطبيق السابع: تصميم برنامج يقوم بادراج الاعداد من ٢٠ حتى ١٠ اى تنازليا من الاكبر الى الاصغر واضافتها فى كائن من النوع jComboBox !!

الكود:

```
jComboBox1.removeAllItems( ) ;
int x ;
x = 20 ;
while (x >= 10)
{
jComboBox1.addItem(x) ;
x-- ;
}
```

موقع الكود:

نكتب الكود السابق فى الكائن(ادرج الارقام).

الشرح:

السطر الاول: يقوم بتفريغ الكائن jComboBox1 من جميع العناصر وقد تم ذلك باستخدام الداله removeAllItems( ).

السطر الثانى: تعريف متغير جديد من النوع int يدعى x .

السطر الثالث: تهيئه المتغير السابق تعريفه بقيمه قدرها 20 .

من السطر الرابع حتى الاخير: حلقه while التكراريه،حيث اذا كانت قيمه x اكبر من او تساوى ١٠ يتم تنفيذ ما بداخل الحلقه والرجوع مره اخرى وتكرارها حتى لا يتحقق هذا الشرط!!

وجسم تلك الحلقه يحتوى على امرين:

الامر الاول: اضافه العناصر الى الكائن jComboBox1 وقد تم ذلك باستخدام الداله addItem( ) .

الامر الثانى: لتكن قيمه x الجديده هى x + 1 .

التصميم:



===========

✓ التطبيق الثامن: تصميم برنامج يقوم باضافه الاعداد الفرديه من ٢٠ حتى ١ وذلك الى كائن من النوع jComboBox !!

الكود:

```
jComboBox1.removeAllItems( ) ;
int x ;
x = 19 ;
do
{
jComboBox1.addItem(x) ;
```

```
x = x – 2 ;
}
while ( x>=1 ) ;
```

موقع الكود:

نكتب الكود السابق فى الكائن(ادرج الارقام).

الشرح:

السطر الاول:يقوم بتفريغ الكائن jComboBox1 من جميع العناصر وقد تم ذلك باستخدام الداله removeAllItems( ) .

السطر الثانى:تعريف متغير جديد من النوع int يدعى x .

السطر الثالث:تهيئه المتغير السابق تعريفه بقيمه قدرها 19 .

من السطر الرابع حتى الاخير:حلقه do ثم while التكراريه،حيث نلاحظ ان هذه الحلقه سوف تنفذ فيها الاوامر التى تلى كلمه do وهى:

```
jComboBox1.addItem(x) ;
x = x – 2 ;
```

ثم نرى الشرط الذى يلى كلمه while وهو:

```
( x>=1 ) ;
```

فاذا تحقق نعود ننفذ ما سبق من اوامر مره اخرى،واذا لم يتحقق نخرج من الحلقه،وبالتالى فان هذه الحلقه سوف تنفذ مره واحده على الاقل.

التصميم:

Item 1

ادرج الارقام

===========

✓ التطبيق التاسع: تصميم برنامج يتضمن كود برمجى يقوم باظهار رساله للمستخدم تخبره بانه لم يدخل القيمه فى كائن من نوع jTextField !!

الكود:

```
if (jTextField1.getText( ).length( ) == 0)
{
JOptionPane.showMessageDialog(null, "please insert the data ?", "worring", 2);
}
```

موقع الكود:

نكتب الكود السابق فى الكائن(تحقق من الادخال).

الشرح:

الداله length( ): تم استخدمناها لعمليه التحقق من وجود نص ام لا داخل الحقل،حيث ان البرنامج يتعرف عليها كمسافه وليس كفراغ.

JOptionPane: مكتبه استدعاء دوال صناديق الحوار.

null: تشير الى عدم وجود اضافات متقدمه.

الداله showMessageDialog( ) : تستخدم فى اظهار الرساله.

بالنسبه للرساله التى سوف يتم اظهارها:

١- محتواها: please insert the data ?

٢- عنوانها: worring

٣- شكل الايقونه: لتحديد الازرار التى نريد ظهورها فى الرساله ولتكن 2 وتشير الى من 0 حتى 3 .

البرنامج:



===========

✓ التطبيق العاشر: تصميم برنامج يتضمن كود برمجى يقوم بعمليه مسح نص مكتوب داخل كائن من نوع jTextField بشرط عند تنفيذ امر المسح تظهر رساله تأكد من عمليه المسح او الغاءه.

الكود:

```
int sign ;
sign = JOptionPane.showConfirmDialog(null, "do you want to delete ?",
"delete", 0,3) ;
if (sign == 0)
{
jTextField1.setText("") ;
}
```

الشرح:

السطر الاول: متغير جديد sign تم تعريفه من النوع int ،يقوم باستقبال القيمه الممرره والتى تمثل ما قام المستخدم بضغطه من الازرار yes او no او cancel .

جمله if: تحتوى على عمليه المقارنه بين القيمه الممرره والقيمه المار التعامل معها كالتالى:

if (sign == 0)

القيم الممرره وما ترمز له:

Yes refer to 0
No refer to 1
Cancel refer to 2

JOptionPane: مكتبه استدعاء دوال صناديق الحوار.

null: تشير الى عدم وجود اضافات متقدمه.

الداله showConfirmDialog( ) : تستخدم فى اظهار الرساله.

بالنسبه للرساله التى سوف يتم اظهارها:

١- محتواها: do you want to delete ?

٢- عنوانها: delete

٣- شكل الايقونه: لتحديد الازرار التى نريد ظهورها فى الرساله ولتكن من 0 حتى 2 .

البرنامج:



===========

تم بحمد الله

الكتاب قيد الاستكمال